



CHRISTIAN-ALBRECHTS-UNIVERSITÄT ZU KIEL

Institut für Informatik, Arbeitsgruppe Theorie der Parallelität

Prof. Dr. K. Jansen, U. Schwarz

26. Oktober 2009

Übungen zur Vorlesung »Effiziente Algorithmen«

Übungsblatt 1

Hausaufgabe 1.1 (\mathcal{O} -Notation und Kostenmaß)

Schreiben Sie zwei RAM-Programme für die Funktion

$$k \mapsto 2^k.$$

Das erste soll die Rekurrenz $2^{k+1} = 2 \cdot 2^k$ ausnutzen, das andere auch $2^{2k} = (2^k)^2$ (*square and multiply*). Bestimmen Sie die Laufzeiten der Programme im Einheitskostenmaß und im logarithmischen Kostenmaß.

Hausaufgabe 1.2 (Laufzeit – Min/Max)

Folgender Algorithmus bestimmt den Index eines maximalen Elements der nichtleeren Zahlenfolge `array` vom Typ `int []`.

```
int maxIndex=0;
for (int i=1; i<array.length; i++)
    if (array[maxIndex] < array[i])
        maxIndex = i;
```

Sei im folgenden N die Länge der Zahlenfolge.

- Bestimmen Sie die Anzahl der von diesem Algorithmus benötigten Vergleiche von Elementen der Zahlenfolge.
- Mit der gleichen Anzahl an Vergleichen läßt sich auch der Index des minimalen Elements der Zahlenfolge bestimmen. Will man nun beide Indizes gleichzeitig bestimmen, erhält man mit dem trivialen Ansatz (beide Algorithmen nacheinander ausführen) eine Laufzeit als Summe der beiden Einzellaufzeiten. Verbessern Sie den Algorithmus derart, dass maximal $\frac{3}{2}N$ Vergleiche von Elementen aus der Zahlenfolge benötigt werden. (Dabei dürfen Sie annehmen, dass N eine gerade Zahl ist.)

Abgabe: Montag, den 02. November, bis spätestens 12 Uhr im Schrein.